

Using the Forest Vegetation Simulator's SVS Keyword and the Stand Visualization System

*Nicholas L. Crookston
RMRS-Moscow
September 27, 1999*

Introduction

A new feature has been added to the Forest Vegetation Simulator (FVS, Wykoff and others 1982) that directly provides data for the Stand Visualization System (SVS, McGaughey 1997; WINSVS, McGaughey 1999). SVS displays a 3D drawing of a stand on a square or a round plot. It has been used widely to display FVS simulation results. The new FVS feature directly outputs SVS tree data eliminating the need to convert detailed FVS tree list files into SVS tree files (FVS2SVS, see the online help in WINSVS). The new feature is faster, does a better job of maintaining the tree location information, and provides more information about the stand in the SVS-generated image than previously possible.

The new feature to FVS builds and maintains a spatially explicit representation of the stand as needed for visualization. There is one tree in the spatially explicit data for each tree per acre in the stand. The spatially explicit data is modified as necessary to account for simulated changes in the stand as stored in FVS's spatially independent data. Data describing standing and down snags is stored and updated to account for decomposition, breakage, and fall-down. Rates used to estimate the changes in snags are partially based on data documented in the Fire and Fuels Extension to FVS (Beukema and others 1999) for the inland northwest United States.

Future enhancements are planned that will result in more realistic images. They may include the ability to display the site-specific cause of tree death, site-specific decomposition rates for standing and down snags, illustration of defoliation and top damage, and the illustration of the effects and behavior of fire. These planned enhancements will result in the creation of images at annual time steps, rather than the current limit of images at FVS cycle intervals of five to 10 years.

The goal in developing the new SVS keyword was to provide visualizations that accurately reflect FVS's behavior and important state variables. Care was taken to insure that the visualization reflects the model and not necessarily reality. Therefore, if the image looks realistic, then perhaps the model is realistic, and conversely, if the image looks unrealistic, perhaps the model needs improvement.

The logic used to create the spatially-explicit data and maintain the relationship between it and the traditional FVS spatially-independent data does not do a perfect job of keeping the two types of data synchronized. The stand in the image should be statistically the same as the stand being projected by FVS. That is, summary estimates of the stand volume, average height, composition of species and sizes, and stand structure classification, should be the same whether computed using the spatially-explicit data (where each sample represents exactly one tree) or the traditional FVS

spatially-independent data (where each sample represents some real number of trees per acre). They should be the same because they both, in theory, represent the same population. However, since statistical sampling is used in generating both the spatially dependent and spatially independent data, there will be differences between estimates computed from the two kinds of data. Furthermore, logic that addresses special cases observed in practice has been included, but the impact of these special cases on the simulations has not been exhaustively tested, and may have unintended consequences on the images produced. Note that using the SVS keyword does not in anyway change the FVS outputs generated when not using this keyword.

Related Programs

The new SVS keyword relies on features found in version 3.28 of WINSVS and subsequent releases.

Suppose (Crookston 1998) version 1.11 contains support for the SVS keyword and the new WINSVS program. However, WINSVS must be installed in the same directory that contains Suppose (generally in C:\fvsbin) if accessed using Suppose "Select Post Processors" window.

Keyword Usage

Use the SVS keyword to generate the input files for the Stand Visualization System. One SVS tree file is generated at the beginning of each FVS cycle and an additional file is generated after any harvesting within cycle s that contain harvesting activities. The keyword follows standard FVS keyword usage conventions.

SVS Turn on the logic that generates the SVS input files.

field 1: Plot geometry and tree placement, where 0=square SVS plot layout while ignoring FVS tree plot designations; 1=square SVS layout and position trees by FVS plot designation, 2=round SVS plot layout while ignoring FVS tree plot designations, and 3=round SVS plot layout and position trees by FVS plot designation (1 is the default).

field 2: Output file organization, where 0=SVS tree data files corresponding to each image are placed in separate files and an index file is created, and 1=all the data are in one file and no index file is created. This format is not currently supported in SVS (default is 0).

The names of the SVS tree data files are automatically generated by FVS. The prefix given to the name of the FVS keyword file name is used when creating the file names. For example, say the name of your keyword file is *run1.key*, the prefix is *run1*. If field 2 is 1 (store data for all images in a single file), then FVS will create a file called *run1.svs* and write all the SVS tree data corresponding to all images to that file. Note that the SVS system can not correctly display the data contained in this file unless it is further processed.

If field 2 is 0 (store SVS tree data in separate files, one corresponding to each image, the default), then FVS will create an index file named *run1_index.svs* (again assuming *run1.key* is the keyword file name), and it will store the SVS tree data in files named *run1_001.svs*,

run1_002.svs, and so on for all the SVS tree data files. If a FVS finds a subdirectory named *run1*, it will store the SVS tree data files in the subdirectory rather than the current directory. FVS always writes the SVS index file in the current directory. Note that the Suppose user interface automatically creates the subdirectory. Also note that recent versions of WINSVS will correctly process the files whether created on UNIX or Windows.

Tree Forms

SVS uses tree form files in addition to the *.svs* files. The SVS tree files created by FVS assume that the following tree classes are defined for each species: tree class 94 represents a snag that has no foliage, class 98 is a recently killed standing snag with red foliage, and class 99 is a normal green tree. Version 3.28 of WINSVS is distributed with the tree form files designed to work with the SVS files directly produced with FVS.

Acknowledgments

I thank Jason Marchinek for programming assistance and Albert R. Stage for providing some great ideas regarding sampling a list of individuals with unequal sampling weights. Consultations with Robert McGaughey were very helpful and always enjoyable. Testing and feedback from Mike Vandyck and Bob Havis from the Forest Management Service Center staff is greatly appreciated.

References

Beukema, S.J., E. Reinhardt, J.A. Greenough, W.A. Kurz, N. Crookston, and D.C.E. Robinson 1999. Fire and fuels extension: model description, working draft. Prepared by ESSA Technologies Ltd., Vancouver, BC for U.S.D.A. Forest Service, Rocky Mountain Research Station, Moscow, ID, 58 pp.

McGaughey, Robert J. 1997. Visualizing forest stand dynamics using the stand visualization system. In: Proceedings of the 1997 ACSM/ASPRS Annual Convention and Exposition; April 7-10, 1997; Seattle, WA. Bethesda, MD: American Society of Photogrammetry and Remote Sensing. 4:248-257.

McGaughey, Robert J. 1999. Stand Visualization System web page:
<http://faculty.washington.edu/mcgoy>

Wykoff, William. R.; Crookston, Nicholas L.; Stage, Albert R. 1982. User's guide to the Stand Prognosis Model. Gen. Tech. Rep. INT-133. Ogden UT: U.S. Department of Agriculture, Forest Service, Intermountain Forest and Range Experiment Station. 112 p.